Zero Knowledge

Anupam Datta or Piotr Mardziel

CMU

Fall 2018
Authentication

What happens when you type in your password?
Naïve authentication

- The server knows your password
- So they can impersonate you at other web sites where you use the same password
“Zero-knowledge” authentication

I know the password
Can you prove it?

acme.com

Can you convince the server that you know your password, without revealing it (or any other information)?
What is knowledge?

What is ignorance?
(lack of knowledge)

• Example 1: Tomorrow’s lottery numbers

2  31  12  7  28  11

We are ignorant of them because they are random
What is ignorance?

• Example 2: A difficult math problem

Prove that $P \neq NP$

We are ignorant because it takes a lot of work to figure out the answer

• Questions of this type include
  – Finding satisfying assignments to Boolean formulas
  – Finding cliques in graphs
  – All NP-hard problems
Using ignorance to our advantage

I know the password
Can you prove it?

acme.com

We want to convince the server that we know the password, while keeping it ignorant of the password itself.

The server is convinced, but gains zero-knowledge!
Goals

• Prover convinces verifier of statement.
  – “I know password for account piotrm”.

• Verifier cannot use what they learned to convince anyone else of the statement.
  – Prevent website from proving they know password for account piotrm.

• *** A record of interaction is not convincing of anything
  – Even for statement: “Piotr knows password for account piotrm”.

Interactive Protocols

Example 1: Non-Color-blindness protocol
Example 2: Cave password protocol
Example 3: Graph coloring
An Interactive Protocol that is Zero Knowledge
A different protocol

I pull at random either two balls from same box or one ball from box 1 and one from box 2

You say “same color” or “different color”

We repeat 10 times

If you got all the answers right, I am convinced you know red from blue

But I did not gain any other knowledge!
Properties

• Soundness
  – If Verifier accepts then the property (Prover is not color blind) holds with high probability

• Completeness
  – If property (Prover is not color blind) holds then the Verifier always accepts
Cave password protocol

- Alice proves to Bob that she knows password.
- Without revealing password to anyone.
Cave password protocol

• Alice proves to Bob that she knows password.
• Without revealing password to anyone.
• A record of interaction does not convince anyone that Alice knows password.
Zero-knowledge

The verifier’s view of the interaction with the prover can be efficiently simulated \textit{without} interacting with the prover

\[ S(V) \approx P \leftrightarrow V \]

Probability distributions on transcripts are indistinguishable
ZK Proof Outline for Non-Color Blindness

• Verifier V*’s view in interaction with Prover P
  1. \(wp \ p\): draw two balls from same box; Prover says “same color”
  2. \(wp (1-p)\): draw one ball from box 1 and one ball from box 2; Prover says “different color”

• \((P,V^*)\) interaction transcript \(\approx S(V^*)\) transcript

  1. When \(V^*\) says “draw two balls from same box” it does so and says “same color” to simulate \(P\) \((wp \ p)\)
  2. When \(V^*\) says “draw one ball from box 1 and one ball from box 2” it does so and says “different color” to simulate \(P\) \((wp \ 1-p)\)
Comments

• Verifier is polynomial time

• Prover has unbounded computation power

• ZK property has to hold for all verifiers $V^*$ (not just the honest verifier $V$)
Brief intermission: cryptographic commitments
Commitments

• Locked box analogy
  – Hiding – hard to tell which message is committed to
  – Binding – there is a unique message corresponding to each commitment
Zero-knowledge password authentication

Oded Goldreich  Silvio Micali  Avi Wigderson

acme.com
Graph coloring

Task: Assign one of 3 colors to the vertices so that no edge has both endpoints of same color

3COL = \{G: G has a valid 3-coloring\}

• Theorem

3COL is NP-complete

• However, it is easy to create 3-colored graphs.
Password authentication via 3-coloring

• Step 0: When you register for the web service, choose your password to be a valid 3-coloring of some (suitable) graph

password:  \[\text{\color{yellow}1} \quad \text{\color{red}2} \quad \text{\color{blue}3} \quad \text{\color{red}4} \quad \text{\color{red}5} \quad \text{\color{blue}6}\]
Password authentication via 3-coloring

- When the server asks for your password

Step 1: do not send the password, but send the graph $G$ instead (without the colors)

password: yellow, red, blue, red, red, blue
Intuition about registration phase

- Because 3-coloring is hard, the server will not be able to figure out your password (coloring) from $G$

- Later, when you try to log in, you will convince the server that you know how to color $G$, without revealing the coloring itself

- The server will be convinced you know your password but remain ignorant about what it is
The login phase, Step 2

password:

You randomly permute the colors

You lock each of the colors in a box and send the boxes to the server

The server chooses an edge at random and asks for the keys to the boxes at the endpoints

You send the requested keys

The server unlocks the two boxes and checks the colors are different

Repeat this 1000 times. Login succeeds if colors always different
Analysis in the login phase

Completeness

If you know the coloring then you will always successfully convince the server
Analysis in the login phase

Soundness

If you are an impostor, you won’t know how to color the graph, so at least one of the edges will have endpoints of the same color.

After $n$ repetitions, the server will fail to catch this with probability (at least)

$$\left(1 - \frac{1}{|E|}\right)^n$$
Analysis in the login phase

Zero Knowledge

If you are honest, the server remains ignorant about your password because all he sees are two random different colors
ZK Proof Outline for 3-COL

• Simulator $S$
  – Internally select random edge $(i, j)$ and random permutation

1. $P \rightarrow V^*$: Generate coloring s.t. color$(i)$ not equal color$(j)$;

   $(P, V^*)$ interaction transcript $\approx S(V^*)$ transcript

   (note by step 1 they are not equal)

Note: If $V^*$ is not honest, use $V^*$ as a blackbox to output edge $e$ in step 2; rewind if $e$ not equal to $(i,j)$
Acknowledgment

- Many slides are from Andrej Bogdanov
- Some slides are from Piotr Mardziel
Seminal Results

• IP and ZK defined [GMR’85]
• ZK for all NP languages [GMW’86]
  – Assuming one way functions exist
• ZK for all of IP [BGGHKMR’88]
  – Everything that can be proven can be proven in ZK assuming one way functions exist